An Investigation into the Performance Evaluation of Connected Vehicle Applications: From Real-World Experiment to Parallel Simulation Paradigm

Presented by
Md Salman Ahmed

Committee Members:
Mohammad A Hoque, Ph.D. (Chair)
Martin Barrett, Ph.D.
Phil Pfeiffer, Ph.D.
Chris Wallace, Ph.D.
Brian Bennett, Ph.D.

EAST TENNESSEE STATE UNIVERSITY
Research Goals

• Four major objectives:

1. Implementation of a connected-vehicle (CV) Application.
2. Devise of a Parallel Simulation Framework.
3. Transportation Network Partitioning.
4. Investigation of the inter-simulator communication overhead.
Research Goal 1
Implementation of a CV Application (freeway merge assistance system)
1st half of 2016 is 10.4% more than 1st half of 2015
Motivation (Cont’d)

About 20-30% of truck accidents happens due to merge conflicts from ramps to freeways (Janson et al.).

Source: Google image
A vehicle on the ramp communicates with the freeway vehicles and comes into a mutual understanding.

- **Creating gap for the ramp vehicle**
- **Slow down for the ramp vehicle**
- **Change the lane if possible**
- **Slowdown or Speedup**
Merge Assistance System

Can be
• Centralized
• Decentralized
• For autonomous vehicle

Scope of this study
Decentralized Algorithms

• Wang et. al. describes a proactive algorithm and suggests speed for all vehicles.

• Park et. al. calculates safe gap using speed and provides both lane-changing & merging advisory information.

• Hayat et. al describes the driver compliance to the advisory messages.
Almost all the merge control algorithms are based on

- Position,
- Speed,
- Acceleration, and
- Time to reach the merging point.
Researchers used different communication techniques for the **merge control algorithms**

- **Infrastructure support such as coded magnets or transponders** (Lu et al.).
- **Internet based** (Wang et al.).
- **Theoretical DSRC** with simulation tools (Park et al., Hayat et al., and Hall et al.).
Most researchers used *simulation tools* to evaluate their algorithms.

No one used actual DSRC devices.

Thus, they overlook some *unanticipated* challenges.
Challenges & Issues (Cont’d)

• Lane detection
  – The merge assistance system should consider only the conflicting vehicles on the rightmost lane.
  – The system should ignore non-conflicting vehicles.

• Advisory start time
  – Each driver should get adequate time to think about the advisory messages.
  – Not too late
  – Not too early
Challenges & Issues (Cont’d)

• Distance up to the merge point

The merge assistance system needs to calculate the distance between the merge point and the current position of a vehicle. But sometimes linear approximation fails.
Fog/Edge computing

- DSRC device may not be able to compute large loads.
- Need to offload some computations (research issue).

Distorted signals

- Signals can be distorted or lost because of the nearby buildings, bridges, steep highways, difference of altitudes etc.
- Need to use extrapolation techniques.

Challenges & Issues (Cont’d)
Challenges & Issues (Cont’d)

• Driver compliance
  – Drivers may or may not comply with the advisory messages.
    – 40% drivers do not comply when the gap sizes are small (Hayat et al.).
    – 32% drivers comply lane changing advisories when they can move to higher speed lanes (Hayat et al.).
The algorithm executes within the ramp vehicle

The basic idea

1. The ramp vehicle keeps track of vehicular trajectories for all the conflicting vehicles on the freeway.
2. Determination of the timing information.
3. Broadcast of control messages.
4. Acknowledge the control messages using synchronization messages.
5. Generation of advisory messages.
6. Display of the advisory messages in the smartphone application.
Assumptions

• We assumed the following criteria:

1. The assistance system is for only the connected-vehicle environment.

2. DSRC communication delay is negligible (order of milliseconds).

3. Computation cost is negligible (order of milliseconds).

4. The entrance ramps are not circular and are not significantly bent.
We collected preliminary data to develop our advisory model.

BSM payload fields:

<table>
<thead>
<tr>
<th>MAC</th>
<th>TS</th>
<th>Lat</th>
<th>Lon</th>
<th>Alt</th>
<th>S</th>
<th>Lat Dir</th>
<th>Lon Dir</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAC</td>
<td>=</td>
<td>Address of the OBU</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TS</td>
<td>=</td>
<td>Timestamp</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lat</td>
<td>=</td>
<td>Latitude</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lon</td>
<td>=</td>
<td>Longitude</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Alt</td>
<td>=</td>
<td>Altitude</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S</td>
<td>=</td>
<td>Speed</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lat Dir</td>
<td>=</td>
<td>Latitude Direction</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lon Dir</td>
<td>=</td>
<td>Longitude Direction</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Location 1

- I-26 (Exit 27, both West and East Bound)
Location 2

• US 321 (The first entrance ramp where University Parkway ends)
Communication Protocol

• Single hop communication protocol for
  – Transmitting BSM packets.
  – Transmitting control messages.
  – Transmitting synchronization messages.

• Bluetooth communication for
  – Displaying advisory message in an Android devices.
Communication Protocol (Cont’d)

- Control messages
- Synchronization messages
- BSM packets
Communication Protocol (Cont’d)

CTRL = Control Message
SYNC = Synchronization Message

Each vehicle is transmitting BSM packets all the time.
• Fields of a CTRL message:

<table>
<thead>
<tr>
<th>MAC</th>
<th>TTC</th>
<th>TDM</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAC</td>
<td>= Address of the OBU</td>
<td></td>
</tr>
<tr>
<td>TTC</td>
<td>= Time to reach the crash/merging point</td>
<td></td>
</tr>
<tr>
<td>TDM</td>
<td>= Timestamp of making advisory decisions</td>
<td></td>
</tr>
</tbody>
</table>

• Fields of a SYNC message:

<table>
<thead>
<tr>
<th>IER</th>
<th>RT</th>
<th>IDM</th>
<th>TTC</th>
</tr>
</thead>
<tbody>
<tr>
<td>IER</td>
<td>= Is the vehicle on the entrance ramp?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RT</td>
<td>= Amount of time spent by a vehicle on the ramp</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IDM</td>
<td>= Is the advisory decision made?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TTC</td>
<td>= Time to reach the crash/merging point</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
1. Approximation of the merging point.
2. Calculation of the average acceleration of the ramp vehicle and average speed of the freeway vehicle.
3. Calculation of the distance between a vehicle and the merging point.
4. Calculation of the time required to reach the merging point using kinematics equations.
1. Calculation of a bearing:

\[ \theta = \arctan2(\sin \Delta \lambda \cos \phi_2, (\cos \phi_1 \sin \phi_2 - \sin \phi_1 \cos \phi_2 \cos \Delta \lambda)) \]  

where \( \phi_1, \lambda_1 \) is the starting point, \( \phi_2, \lambda_2 \) the end point and \( \Delta \lambda \) is the difference in longitude.

2. Predicting the merging point:

\[ \delta_{12} = 2 \arcsin\left(\frac{\sin^2\left(\frac{\Delta \phi}{2}\right) + \cos \phi_1 \cos \phi_2 \sin^2\left(\frac{\Delta \lambda}{2}\right)}{2}\right) \]

\[ \theta_a = \arccos(\sin \phi_2 - \frac{\sin \phi_1 \cos \delta_{12}}{\sin \delta_{12} \cos \phi_1}) \]

\[ \theta_b = \arccos(\sin \phi_1 - \frac{\sin \phi_2 \cos \delta_{12}}{\sin \delta_{12} \cos \phi_2}) \]

if \( \sin(\lambda_2 - \lambda_1) > 0 \)

\[ \theta_{12} = \theta_a \]

\[ \theta_{21} = 2\pi - \theta_b \]

else

\[ \theta_{12} = 2\pi - \theta_a \]

\[ \theta_{21} = \theta_b \]

\[ \alpha_1 = (\theta_{13} - \theta_{12} + \pi)\%2\pi - \pi \]

\[ \alpha_2 = (\theta_{21} - \theta_{23} + \pi)\%2\pi - \pi \]

\[ \alpha_3 = \arccos(-\cos \alpha_1 \cos \alpha_2 + \sin \alpha_1 \sin \alpha_2 \cos \delta_{12}) \]

\[ \delta_{13} = \arctan2(\sin \delta_{12} \sin \alpha_1 \sin \alpha_2, \cos \alpha_2 + \cos \alpha_1 \cos \alpha_3) \]

\[ \phi_3 = \arcsin(\sin \phi_1 \cos \delta_{13} + \cos \phi_1 \sin \delta_{13} \cos \theta_{13}) \]

\[ \Delta \lambda_{13} = \arctan2(\sin \theta_{13} \sin \delta_{13} \cos \phi_1, \cos \delta_{13} - \sin \phi_1 \sin \phi_3) \]

\[ \lambda_3 = (\lambda_1 + \Delta \lambda_{13} + \pi)\%2\pi - \pi \]

3. Calculation of a linear distance:

**Haversine Formula**

\[ a = \sin^2\left(\frac{\Delta \phi}{2}\right) + \cos \phi_1 \cos \phi_2 \sin^2\left(\frac{\Delta \lambda}{2}\right) \]

\[ c = 2 \arctan2(\sqrt{a}, \sqrt{1-a}) \]

\[ d = R \cdot c \]

4. Calculation of the timing:

\[ d = u \cdot t + \frac{1}{2} a \cdot t^2 \]

\[ d = v \cdot t \]
Advisory Messages

• All vehicles receive and synchronize the timing information using CTRL & SYNC messages.
• Then each vehicle generates advisory messages for itself.
• Four different advisory messages used:
  – Notify the presence of a merging vehicle on ramp
  – Advise the driver to maintain the speed
  – Advise the driver to merge behind a specific vehicle
  – Advise the driver to slow down to allow merging in front
Evaluation of the Model

• Evaluated our model for 8 exits in I-26
  – Exit 27 West Bound & East Bound
  – Exit 32 West Bound & East Bound
  – Exit 34 West Bound & East Bound
  – Exit 36 West Bound & East Bound

• Experimental Setup inside vehicle
Advisory & Alert Messages in the Android App (VIDEO 1)
Advisory & Alert Messages in the Android App (VIDEO 2)
Preliminary Data Analysis

Exit 27 (East Bound)

Exit 27 (West Bound)
Result Analysis

Average speed
vehicle 1: 52.32 mph
vehicle 1: 52.20 mph

Average acceleration
1.155 m/s ^ 2
Or
(2.58 mph/s)
Future Plans

• Consideration of circular ramps.

• Lane detection mechanism.

• Discarding mechanism for non-conflicting vehicles.

• Automatic cruise control feature.
Research Goal 2
A Parallel Simulation Framework
## Background

<table>
<thead>
<tr>
<th>Simulators</th>
<th>Two simulation models of a simulator</th>
<th>Network model</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASH</td>
<td>IDM/MOBIL, IVG</td>
<td>SWANS</td>
</tr>
<tr>
<td>OVNIS</td>
<td>SUMO</td>
<td>NS-3</td>
</tr>
<tr>
<td>STRAW</td>
<td>Developed their own model</td>
<td>SWANS</td>
</tr>
<tr>
<td>Veins</td>
<td>SUMO, IVC</td>
<td>OMNET++</td>
</tr>
<tr>
<td>VnetIntSim</td>
<td>INTEGRATION</td>
<td>OPNET</td>
</tr>
<tr>
<td>TraNS</td>
<td>SUMO</td>
<td>NS-2</td>
</tr>
<tr>
<td>iTETRIS</td>
<td>SUMO</td>
<td>NS-3</td>
</tr>
<tr>
<td>GrooveSim</td>
<td>Developed their own model</td>
<td>Their own network model</td>
</tr>
<tr>
<td>Automesh</td>
<td>Customizable to add any mobility model</td>
<td>NS-2 or Qualnet</td>
</tr>
</tbody>
</table>
Challenges of a Parallel Simulator

- Partitioning of the Complex Transportation Network.
- Reducing Inter-Simulator Communication Overhead.
- Existence of heterogeneous vehicles.
- Synchronization problem.
- Scalability of Parallel Simulation.

Scope of this thesis
Research Goal 3
Transportation Network Partition
Network Partitioning: Parameters

• Traffic parameters extracted from openstreetmap.org (OSM data)
  – Node weight
  – Link Length
  – Number of lanes
  – Link density (approximated)
  – Link priority
Experimenting with METIS

For the road network of Johnson City, TN.

<table>
<thead>
<tr>
<th># partition</th>
<th>Communication cost (given by METIS)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Without traffic parameters</td>
</tr>
<tr>
<td>2</td>
<td>10</td>
</tr>
<tr>
<td>3</td>
<td>13</td>
</tr>
<tr>
<td>4</td>
<td>22</td>
</tr>
<tr>
<td>5</td>
<td>23</td>
</tr>
<tr>
<td>6</td>
<td>30</td>
</tr>
</tbody>
</table>

N_{adj[v]} be the number of domains other than P[v]
P be a vector of size |V|
Experimenting with METIS (Cont’d)

Without traffic parameters

With traffic parameters
Research Goal 4
Investigation of the inter-simulator communication overhead
A linear partial results accumulation algorithm for a distributed matrix-matrix multiplication (Hoque et al.).

Each process sends and receives partial results from other processes.

Communication Overhead: \((p-1)(z^2N/p + \delta)\)
Logarithmic Algorithm: Summary

Communication Overhead

$$\log_2(p)(z^2N/p + \delta)$$
Performance Analysis (Cont’d)

Total communication overhead for N = 1M

- Original Pattern
- Logarithmic Pattern

Time (sec)

Number of processes

0 2 4 8 16 32 64 128 256 512

0 1 2 3 4 5 6 7 8 x 10^4
Concluding Remarks

• Improvement of the freeway merge assistance system
• Development of the simulation framework
• Improvement of the partitioning scheme


5. Md Salman Ahmed, Jennifer Houser, Mohammad A Hoque, Phil Pfeiffer, Reducing Inter-Process Communication Overhead in Parallel Sparse Matrix-Matrix Multiplication, ACM Mid-Southeast Conference 2016. (Best paper award, Masters Category, third prize)


